

# Q U A N T A

## THE NEWSLETTER OF THE INDEPENDENT QL USERS' GROUP

*Note: August 2014 - In enhancing the original copy of this issue, it has been found necessary to change the font sizes considerably from the original and thus a number of articles do not start on the page that they did in the original. As a result of these changes, in the Contents List below where two pages are given viz: 18 - (19), the first is the page number in the original and the second, bracketed, is the page number in the .pdf web version.*

VOLUME 1 ISSUE 11

DECEMBER 1984

### CONTENTS

- 2. Information on the Group
- 3. Editorial
- 3. Members' letters
- 6. Problem section
- 8. Review of CST Q- $\pi$  parallel interface
- 9 - (8). Connecting the Brother M-1009
- 10. Metacomco QL Assembler - Ackermann function
- 12 - (13). IQLUG Program Library
- 15. List of IQLUG Sub-Librarians
- 15 - (16). QL Computing by Ian Sinclair
- 16 - (17). Notes on ComPec, Olympia
- 17. The ZX-Microfair
- 17 - (18). Shinwa CP-80 type 1 printer
- 18 - (19). Easel graph printing to Brother HR-5 printer
- 19 - (20). Spectrum as spooler for QL
- 20 - (21). The TAB function on the QL
- 20 - (21). Recovering crashed Quill documents
- 21. QL Qarrying Qases
- 21. Hardware review: Prism QL 14 colour monitor
- 22 - (23). Settings for the Brother EP 44 printer
- 23 - (24). Recursive functions on the QL
- 24 - (25). Archive limitations
- 26. Zero Electronics print buffer
- 27. Microline 82A printer
- 27 - (28). Pascal review
- 31 - (32). Using serial interface to connect BBC to QL
- 33. More on the Epson RX-80 switch settings
- 33 - (34). Using Quill
- 34. QLMON
- 34 - (35). QLkit announced
- 35. Metacomco BCPL compiler
- 36. Case conversion program written in BCPL
- 37 - (38). Beginners section
- 39. Birmingham area sub-group

## INFORMATION ON THE GROUP

Membership of the Group is by subscription to Quanta, the Group's newsletter, which is published monthly. Membership details are obtainable from the Group Secretary. Membership of the Group is open to anyone with an interest in the Sinclair QL microcomputer.

Members requiring assistance with problems related to the QL may call the Secretary. An attempt will be made to put them in touch with a member who can help with the problem.

Workshops will be arranged from time to time in various parts of the country.

The Group maintains a software library. Most of the programs are free to members. The library list is published from time to time in the newsletter.

A membership list is obtainable from the Secretary.

Please send all contributions for the Newsletter to the Editor.

Chairman

Leon Heller,  
8 Morris Walk, Newport Pagnell,  
Bucks. MK 16 8QD.  
Tel: (0908) 613004

Software Library:-

Rob Sherratt,  
'Betheden',  
129 Melford Way,  
Cavendish Park,  
Felixstowe,  
Suffolk IP11 8UH.  
Tel: (0394) 272782

Secretary and and Newsletter Editor:-  
Newsletter Publisher:-

Brian Pain,  
24 Oxford Street,  
Stony Stratford,  
Milton Keynes.  
Tel:- (0908) 564271

SINCLAIR, QL, QDOS, QLUB and ZX Microdrive are Trade Marks of SINCLAIR Research Ltd.

Quill, Abacus, Archive and Easel are Trade Marks of Psion Ltd.

## EDITORIAL

We now have sufficient members and sufficient contributions, together with one or two advertisers, to justify an increase in size, so this issue comprises 40 pages.

The Swindon workshop was a great success, with an attendance of over 150. For a relatively new group like ours this is extremely encouraging. Feedback on the weekend from members has been very favourable, and we will be holding another one next spring.

Another success story is the software library, which continues to expand. Three cartridges are available, and Rob Sherratt tells me that he is now on the fifth cartridge, so we must have over 100 programs, many of very high quality, in the library. From now on, each issue of the newsletter will contain a library coupon on one of the pages. If you order anything from the library, you must submit the coupon from the current issue of the newsletter. It is double-sided so it cannot be easily photocopied. This will ensure that only paid-up members of IQLUG use the library facilities. This was Rob's idea, and means that the software librarians do not have to mess about with membership lists and membership numbers. Talking about the software librarians, members MUST get library software from them, not from Rob, who is now only responsible for library admin., Not software distribution. Due to the size of the library list, it will be published separately from the newsletter, at a cost of £1.50. This will enable us to improve the library service, and provide librarians with sufficient cartridges for holding the library and backup copies. If anyone has any problems with the library, such as inordinate delays, please ring the appropriate librarian in the first instance (Brian Pain can give you the phone number), and/or contact Rob.

My "battle" with Sinclair over the Psion upgrades is still continuing. To provide me with some ammunition, could any members who are having serious problems with shortcomings and bugs in the Psion packages, and who do not agree that the Psion software completely outperforms existing software for micros, please drop me a line detailing the difficulties they are encountering, with the version number of the package. I will then pass the information to Sinclair for comment.

Leon Heller

## MEMBERS' LETTERS

The membership of IQLUG seems to fall pretty much into two groups, program users and program writers and the difference between them is a good bit larger than you're prepared to admit. At the moment the program writers are getting more out of Quanta than the users. Users after all can be pretty illiterate in computer terms and still successfully use the Psion packages for example, so they are unlikely to be able to write anything for the newsletter.

I've experienced similar effects to those reported by Robin Smith of Aberdeen. The 'quiet crash' in which the cursor disappears and the machine does not respond to the keyboard. The double F2 Press does not cure this. Also a more drastic one in which the screen adopts a jagged random pattern then goes blank while the mdv2 goes haywire. On attempting to get re-started the Quill tape is not accepted by the machine, and the cartridge from mdv2 proves to be wiped clean, or rather, dirty, since the machine now refuses to even format it. After two like this I sent computer and master tapes back to Sinclair, who replaced both. I received a D0-9 in place of a D0-7, and a Quill 1.03 ACB in place of a 1.03 ADB (any difference?). With my replacement set I have experienced one 'quiet crash'. Perhaps someone could devise a program to recover the contents of the Quill edit file called 'def\_doc', in the event of one of these 'quiet crashes'.

Some notes in Quanta on the possibilities of disc drives would also be appreciated. I guess that, cost apart, it would be better to standardise on disks than on cartridges.

Keep up the good work,

Ron Bezzant,  
24 North Grange Road,  
Bearsden,  
Glasgow. G61 3AF.

-----

Membership - Advertise in Popular Computing Weekly. The mag. has consistently supported the QL since launch and has had a QL page for many months - so can be presumed to have quite a lot of QL owners in its 56.000 readers. /x We are! x/

MDV1 noise/failure - This must be a mechanical fault. I received my D05-001727 back as D12--- with JM installed in late October and found mdv1 would not read protected cartridges at all - either giving 'not found' messages or 'bad medium's so off it has gone to Camberley again.

Postage refund - On reporting the above to Camberley, Sharon who answered the phone insisted that Sinclair would refund my postage. I'll let you know if they don't.

QL Users Manual - Do all members know that if you got the original QL Users Manual then a phone call to Camberley will get you the 2nd edition - with 6/84 at the bottom of most of the Pages.

Psion help - They are QUICK - but that's all I can say in their favour. Answers are facile - ie there is a bug but tough until a later release - or are surprising - 'It is not good practice to write large documents when using a word processor because there is always a danger that you might loose (sic) Your work - Funny, I thought a word processor was FOR big documents. At present I am finding bugs faster than I can write to Psion - and I haven't even started on Abacus/Archive yet!

Pipelining - The article 'An Expensive Alternative' in Oct/Nov QL User stated that 68K/OS had a PIPE: device, which exists within QDOS but is not advertised by Sinclair. No further reference to how to access the QDOS PIPES has been seen. Now I know it exists - I have seen my FB QL read a Microdrive file without touching it (second RUN around) and likewise DELETE files without spinning the tape (bit more dubious) but I cannot duplicate with my JM QL.

Gerard Phelan,  
17 Gunnersbury Court,  
Bollo Lane,  
London W3 8JL.

/x PIPES can be accessed with Tony Tebby's QLkit, mentioned in this issue. Your experiences with files loading or deleting immediately are nothing to do with PIPES, but result from the large buffers used by QDOS, I think. x/

You may well notice that I have managed to interface the QL to Brother HR15 daisywheel printer, on my own initiative, by the simple expedient of thoroughly reading the two Manuals. This, together with a soldering iron to connect the number 6 pin to the number 8 Pin on the cable, brought the printer to life.

Ken Brookes,  
3 Herontye Drive,  
East Grinstead,  
West Sussex RH19 4LR.

-----

As far as my CM 14 colour monitor is concerned (?= the 1400 asked about in Issue 10) I had it modified in the shop where I bought it and it shows 84 characters on the screen, losing one on the r.h. edge. It cost me £199 and performs very well.

John Roberts,  
19 White Horses Way,  
Littlehampton,  
West Sussex BN17 6NH.  
(09064) 4405.

-----

I am going to attempt reading from a tape recorder via the Network connector. However, I don't expect much success.

/x I cannot decipher this member's signature. Has anyone else tried this? x/

You were good enough to publish a letter of mine in issue 9 of Quanta (page 10), in which I sought help and advice with a programming problem. Two members were kind enough to write with suggestions, Capt.E.D.Fieldson of Rickmansworth, Herts. & Mr.J.Foley of Shrewsbury, Shropshire. Their respective solutions to my problem are as follows:-

```
a)  1 DEFine PROC L(a,b)
      2 IF a AND b THEN LIST a TO b
      3 IF NOT a AND b THEN LIST TO b
      4 IF a AND NOT b THEN LIST a TO
      5 IF NOT a AND NOT b THEN LIST
      6 END DEFine
```

The NOT a or NOT b condition is met by entering a 0 as the appropriate parameter.

```
b)  1 DEFine PROC LL
      2 INPUT#0," LIST FROM LINE!!aa$!!"TO LINE!!Bb$
      3 IF aa$=" " AND bb$=" " THEN LIST
      4 IF aa$<>" " AND bb$=" " THEN LIST aa$ TO
      5 IF aa$=" " AND bb$<>" " THEN LIST TO bb$
      6 IF aa$<>" " AND bb$<>" " THEN aa$=aa$-10:LISTEM
      7 END DEFine
      8 DEFine PROC LISTEM
      9 REPEAT PRINTEM
      10 aa$=aa$+10
      11 LIST aa$:IF aa$=bb$ THEN EXIT PRINTEM
      12 END REPEAT PRINTEM
      13 END DEFine
```

Lines 6 & 10 assume that program lines are equally spaced at intervals of 10.

Both these suggestions achieve the desired result though neither member was able to suggest what was wrong with my original listing (other than confirming that it did not work). I should like to express my thanks for their efforts. Capt. Fieldson's letter arrived first and I incorporated his suggestion in a program of utilities which I submitted to the library.

I should like to make two other points

#### Saving part of a spreadsheet:

I was using Abacus to do some repetitive calculations with different sets of data and wanted to store the results. It seemed wasteful to save the entire spreadsheet each

time and I could find no mention of how to save only the cells containing results. I decided to export the result cells to Archive. This I have so far failed to do (is this another bug or have I just not read the instructions properly?). In frustration I tried to import back to Abacus and found that this worked fine. Now I have a copy of my spreadsheet saved and also a copy of the results of each individual calculation. The results can be displayed in their original setting or else all the results can be viewed side by side in an empty spreadsheet. Happiness!

### Multitasking from BASIC

Was this ever promised or was it all a misunderstanding? We have an IBM PC in work and have recently got it to multitask using some software called TASCMASTER. This allows multitasking from BASIC by loading BASIC from disc to two (or more) separate memory segments but even just two BASIC programs running simultaneously have a disastrous effect on each other. It's like using a Spectrum or worse. Machine code seems to be the thing.

Keep up the good work.

David Guthrie,  
2 Hillside Drive,  
Stranmillis,  
Belfast. BT9 5EJ.

### **PROBLEM SECTION**

I currently understand very little about programming, would someone please suggest suitable material I can study or other course of action which can be undertaken to obtain understanding and proficiency.

A Morton,  
Bungay Road,  
Brooke,  
Norfolk. NR15 1DX.

/x The best book I have yet come across for learning SuperBASIC is 'BASIC Programming on the QL' by Neil and Pat Cryer, published by Prentice/Hall at £7.95. x/

After an initial sticky patch, when my copies of the bundled software kept crashing with "bad medium" errors and the Easel tape failed and had to be returned, (it was replaced in 10 days), I have been very impressed, especially with Abacus. However I cannot see the system being useable in my work, which involves following the surgical behaviour of upwards of one thousand patients' eyes, without proper fast disc based storage. The sooner we learn how to interface the QL to our existing disc drives the better! I have a pair of 80-BUS QDSS drives and would much value advice on how to use these with my QL. Quite apart from professional use, the slow access time and holdups while using Quill are a real nuisance. I hope the next software upgrade can get more of a file into RAM, especially so that decent use can be made of Abacus as well.

While on holiday in the south of France (it rained), I found MDV2 turning on spontaneously after 15-20mins use. I presume this was the heating problem mentioned by you in the newsletter, for it has yet to recur in the cold Northwest. Incidentally, I do not recommend anyone to try any serious work on their Micro in France...I was organising material for a paper to be given at a meeting on my return when, after a whole morning's work, there was a glitch in the mains power supply.....Thankfully red wine was at hand! In future I SAVE every 10 mins....

I would also like to hear of any plans to interface graphics tablets. One project in which I am involved concerns the analysis of monolayers of polygonal cells lining the inner surface of the cornea of the human eye. These can be photographed in life by

specular microscopy, and at present we calculate individual cell sizes by vector analysis of projected negatives viewed on a CCTV system using a Magiscan-1 minicomputer and then analyse the results statistically with a Fortran program..... Life would be considerably simpler if we could do the job on site, on a micro, and projection of the negs onto a graphics tablet could provide a simple portable alternative.

Finally, there remains a dearth of suitable software examples in SuperBASIC for the non-hacking rank amateur such as I. Up till now, I have been playing about with Nascom Microsoft Basic and the Polydos enhancements, and SuperBASIC is both a revelation and something of a mystery as yet. I am sure many would appreciate help in this direction.

Keep going !

A.E.A.RIDGWAY,F.R.C.S.,  
10 FRAMINGHAM RD.,  
BROOKLANDS,  
SALE, CHESHIRE.  
M33 3SG

/x Some graphics tablets have an RS-232 interface, so all that is required is suitable software. x/

Among other things, I have a working program which enables me to communicate with a CPM system (on an ancient North Star Horizon), Keep all my QL files on floppy disc, and down-line load (or merge) them into the QL. Of course, I have to use a micro-drive stored boot-program to boot the rest of the files to the QL.

I have uncovered some bugs in the course of this, most of which I have been able to program around. (Has anyone yet coined the phrase "buggering around" for doing that?) But there are some which defeat me. The most annoying is that I cannot get:

SAVE ser2

to work, (all parameters tried), in spite of every success with:

LOAD ser2hc

and

MERGE ser2hc

so I have to save files to Microdrive, and then transfer them to CPM floppy disc, reading them line by line.

If anyone else has any relevant experience I should very much like to contact them.

Lawrie Moore,  
58 Clare Court,  
Judd Street,  
LONDON  
WC1H 9QW.

Stop press!

I have just been talking to Martin Baines at CST, and he has promised to send one of their disk interfaces to me in a few days time. A full review will be in the next issue.

Leon Heller

## REVIEW OF CST Q- $\pi$ PARALLEL INTERFACE

The Q- $\pi$  produced by Cambridge Systems Technology is a Centronics compatible parallel port for the QL. Since it contains its own driver software in ROM, and is treated as an output device called 'Par', it is very easy to use, and offers considerably more than the serial to parallel converters such as that sold by Miracle Systems.

The unit consists of a small PCB, mounted in a black plastic box which matches the QL, with a 64-way connector on one end, and a 34-way connector on the other end. It comes with a ribbon cable about 2 m in length, one end of which connects to the Q- $\pi$ , the other end carries a Centronics type connector for attachment to the printer. The unit looks very well made, and appears to be buffered, with its own power supply, derived from the QL bus, so the electrical design looks excellent. Only 10 pages of documentation are provided, but it includes everything you need to know, and is very easy to follow, with plenty of examples.

The device is connected to the QL by first switching off the power, and then plugging it into the expansion port on the left of the machine. The user is warned against connecting or disconnecting the unit with the power on, which the documentation says might damage the interface, the QL, or both! It doesn't tell you what to do if such damage takes place. I forgot once and removed the Q- $\pi$  with the power on, but nothing untoward happened, although it is not to be recommended. The printer may then be attached and the QL Powered up. The usual initial display results, but at the top of the screen the Message 'CST Q- $\pi$  Printer Module V 1.02 c 1984' is displayed, indicating that the system recognises the interface. On selecting F1 or F2 things appear as normal, except that you now have an additional output device called 'Par', which behaves just like any other device such as the screen or the RS-232 Ports, enabling the user to type things like

```
copy mdv2_fred_bas to par_3000
```

The number after the device name is the size of an optional buffer, so if your file fits into the buffer, you can carry on using the computer while the printer gets on with it, like a spooler.

I used the Q- $\pi$  with a borrowed FX-80 printer, and it worked very well. I won't go into all the features provided by the unit, except to say that a screen dump facility is provided (Just type `dump_scr <channel>`) which is very useful. It also works very well with the Psion packages. It is also designed to plug into the forthcoming Sinclair expansion unit, and may be allocated a SLOT number in the expansion module.

To sum up, this is an excellent unit and good value for money at £49.99, including cable, VAT and carriage. CST will be offering a 5% discount to IQLUG members who order direct from them at:-

Cambridge Systems Technology,  
30 Regent Street,  
Cambridge.  
Tel: (0223) 323302.

CST also do an HP-IB or IEEE interface for the QL, which will be of interest to users wishing to attach their machine to Hewlett-Packard instruments and other HP-IB compatible equipment.

## CONNECTING THE BROTHER M-1009

The reason I am writing is to help add to your data base of How To Connect Things To The QL. The two items (which I have not seen mentioned yet) are the Ferguson TX with RGB telly monitor (£7.95 a month from Radio Rentals) and the Brother M-1009 dot matrix printer (£199.95 from Menzies [or W.H. Smiths for those south of the border]).



The printer driver:

```

DRIVER NAME      : BRO M-1009
PARITY           : EVEN
BAUD RATE       : 9600
LINES/PAGE      : 66
CHARACTERS/LINE : 80
CONTINUOUS FORMS : YES
END OF LINE CODE : <CR>,<LF>
END OF PAGE CODE : <FF>
PREAMBLE CODE   : <ESC>,@
POSTAMBLE CODE  : NONE
EMPHASIZE ON    : <ESC>,E
EMPHASIZE OFF   : <ESC>,F
UNDERLINE ON    : <ESC>,-,1
UNDERLINE OFF   : <ESC>,-,0
SUBSCRIPT ON    : <ESC>,S,1
SUBSCRIPT OFF   : <ESC>,T
SUPERSCRIP ON  : <ESC>,S,0
SUPERSCRIP OFF : <ESC>,T
TRANSLATE 1    : £,£
TRANSLATE 2    : NONE

```

Consult the manual to find out how to make these changes (for those who still haven't found it, the instructions on how to set things up are in the "Information" section at the back of the manual) because it's fairly tricky. It's easiest to do with a "NEW COPY" (F1) of the FX-80 template.

The RS-232-C cable provided free(?) to those lucky(?!!) enough to receive their QL late, but functional, plugs into Ser1 and the RS-232 socket on the back of the Brother (be sure and insist on a dual standard model, some come supplied with a Centronics interface only). The only thing to do then is set the dip switches inside the printer to the proper position.

I just discovered that trying to delete a line with [CTRL][DOWN ARROW] crashes Quill 1.03 ACB, corrupting both cartridges, rendering Quill unusable and destroying any possibility of access to whatever files used to be on mdv2\_. I am afraid that it also destroyed my working copy of this letter to you, Leon, so rather than sending you this on a cartridge as I had originally intended, I will send you a letter because I already have page 1 printed and I've no time to retype it in.

Back to the dip switches: after consulting the Brother's manual to find out which set of switches are which, set them thus (following the manual's convention):

2.1	ON	1.1	OFF
2.2	ON	1.2	OFF
2.3	OFF	1.3	ON
2.4	OFF	1.4	ON
2.5	OFF	1.5	ON
2.6	OFF	1.6	OFF
2.7	ON	1.7	ON
2.8	ON	1.8	OFF

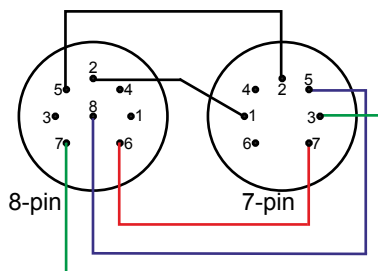
With the above arrangements made, everything plugged in and turned on and the printer in "on-line" status, printing from Quill should be as easy as F3,P,ENTER,ENTER,ENTER. Well, it worked for me anyway (you're looking at three day's fiddling).

Now to the monitor. The Fidelity RGB is available on rental from Radio Rentals on Prince's Street in Edinburgh, and thus I assume nationally from the same company. It's

big feature is the provision of separate RGB (7-pin DIN) and video (5-pin DIN) for those of you who intend to use the QL for subtitling all your home movies) sockets on the back selected by channels 7 and 8 on the front (they can be tuned normally for TV if there is nothing in the socket for all of you looking forward to that time when we shall have more channels). I finally managed to wire up a cable that works, details to follow, and am pleased with the display. It is by no means up to the standards of the Cub™ dedicated monitor, but the image is rock steady with none of the video crawl that was so annoying when using the aerial socket. Also it allows easy switching between sources without bulky outboard boxes. I only get 80 columns on it, but by suitably configuring the margins after booting, with F1 I find I get a useable display, fully readable, with only the loss of two letters per line in the command window. Perhaps if I owned it and knew enough about these things I could fiddle around inside and fix things even better. I am pleased with the combination (except when I compare it to a Cub™).

The lead requires five-core cable, a 7-pin DIN and an 8-pin DIN plug (in Edinburgh all available from Brown's on the Bridge). Pin connections are as follows:

8-pin Sinclair		7-pin Ferguson
1		1 GND
2	GND	2 VSYNC
3		3 GREEN
4		4
5	VSYNC	5 BLUE
6	RED	6
7	GREEN	7 RED
8	BLUE	



I am including a diagram to help.

Fred Holliss,  
11 London Street,  
Edinburgh. EH3 6LZ.

### METACOMCO\_QL\_ASSEMBLER - Ackermann\_Function

I was very anxious to make a start with 68008 Assembly programming, and was very pleased to receive the Metacomco Assembler two weeks ago. This article is not intended to give a real assessment of this assembler: I feel that I would need to use it over a longer period - but my first impressions are that it is excellent: the editor is very comprehensive and easy to use, and the error codes are almost uncannily detailed - so much so that I half expected it at times to say that the error was a missing inverted comma, and that it had corrected it for me! Perhaps the next generation of assemblers will correct mistakes, or offer a selection of corrections for one to choose from!

Needless to say, I looked back with sadness to the absence of the familiar Z80 instructions - the vigorous 'PUSH' and 'POP' replaced by the wordy but sedate 'MOVE.L (D0),-(SP) and 'MOVE.L (SP)+,D0'. 'INC's and 'DEC's were so explicit. And 'LDIR' and its cousins almost majestic in their power. It was only very slowly, after using the assembler for some time, that the advantages of the 68000 code became apparent, and seven A and seven D registers are real wealth! The manual provides three very good programs with increasing complexity to copy out, and it is amazing how one can make mistakes with unfamiliar code even when doing this simple task. Assembly is time consuming, and I soon learnt to use defaults and not to ask for a 'list' file and machine code file until virtually all errors were eliminated. But of course a listed file is invaluable at some point to understand what is going on.

However, the plunge into writing 68000 code, must be made and I decided to take Leon Heller's program 'Ackermann\_Function' (QUANTA July p24) to work on. This needs all the first section to be replaced by QL compatible code since the six calls

CHROUT to WSTART are of course inapplicable to the QL. But the mathematical part, and most of the binary conversion (which is of permanent value as a sub-routine), can be left intact. Having bought a copy of Tony Tebby's book on QDOS V1.03 routines, the idea of 'traps' had been digested - at any rate, in part! Hence, the first item is to create a 'job' and 'open a channel', and provide means of closing and tidying up - procedures involving traps 1 and 2, fortunately given in the examples in the METACOMCO manual. The programme is self-locating in memory, i.e., the code is position independent. The actual location (in the 3F0000 to 40000 region) can be found by:

```
10 MEMLOC = RESPR(444)
20 PRINT MEMLOC
```

Trap 3 involves input and output, and the simplest example of this is the IO\_ FBYTE (EQU 1) and IO\_ SBYTE (EQU 5). The position is, that provided D0 is made equal to 1, the line 'TRAP 3' causes a byte to be fetched from the screen: similarly if D0=5, TRAP 3 causes a byte to be printed on the screen. Somewhat more complicated are IO\_ FSTRG (D0=3), IO\_ FLIN (D0=2) and IO\_ SSTRG (D0=7) which require D1 and D2 to give the length and address of the string in question. There must be some 70 or more different IO codes covering location on the screen, colour, loading and saving a file, and every possible operation relating to input and output. There is, of course, no need to use such lengthy expressions as IO\_ SSTRG, IO\_ FBYTE, etc. For the sake of standardisation I have used them here, but have contracted them to Ssg, Fbt, Sbt, Flin, etc in my own files. I cannot understand why there is such love for the underline '\_' and lengthy expressions which always slow up typing and use up memory!

I was very pleased to see MACROS can be used in this Assembler and have therefore used them in this program wherever possible. The first two are concerned with TRAPS and finishing. The third, QUAN1 ('Question/Answer 1'), puts questions and receives bytes entered on the keyboard as answers. The second, QUAN2, merely prints headings or statements. To make the Macros of general application, I have written the strings in sequence at the end as MS1, MS2, MS3, etc so that MS1 is the address of the first string and (MS2-MS1) its length: similarly MS4 and (MS5-MS4) for the fourth. This avoids the need to count the length of messages, and makes it possible to alter any message at will without altering the programme.

Note that m is temporarily stored in D6 and n in D5 - the more normal method would perhaps be to push them on to the stack and retrieve them after both Macros are used - but one feels positively mean in not using any of the registers beyond D3! The range of figures entered by the single byte is extended beyond 9 by using A, B, C, ..., Z, ..., z but the Ackermann\_function needs a great number of reiterations, and Ack(3,4) gives an answer of 125: 3,5 253: 3,6 509: 3,7 1021: 3,8 2045. But it took nearly 3 minutes wait for 2045 and I got tired of waiting for Ack(3,9). Thus there is little use for high values of m and n.

As mentioned earlier, little alteration is needed to the maths - simply to replace the print-out command JSR CHROUT in BINDEC2 by the Macro QDOS IO\_ SBYTE,3, and to pop the character off the stack into D1 (not D0 which here is used by the TRAP) [MOVE.B(SP)+,D1] and use D2 for counting the loops.

I was dismayed to find that the 444 bytes used represented 36% 'workspace used', and the limitation at present is in size of programme. I suppose that this refers to space on the cartridge, so the use of an otherwise empty cartridge would improve matters. Presumably, this is not wholly related to the choice of assembler, and is a general problem.

With the METACOMCO assembler it is possible to work on two different programmes in two different windows. My question is why on earth one would want to do this!! But I can see the possibility of writing one programme and loading in another with various subroutines or programmes for consultation and possible 'import'. This could in some cases save a good deal of time during assembly.

**\*\*\* ACKERMANN FUNCTION \*\*\***

QDOS Request Codes

```
MT_FRJOB EQU $05
IO_OPEN  EQU $01
IO_CLOSE EQU $02
IO_FLINE EQU $02
IO
```

**\*\*\* ACKERMANN FUNCTION \*\*\***

QDOS Request Codes

```
MT_FRJOB EQU $05
IO_OPEN  EQU $01
IO_CLOSE EQU $02
IO_FLINE EQU $02
IO_SSTR  EQU $07
IO_SBYTE EQU $05
IO_FBYTE EQU $01
```

**\*\*\* MACROS \*\*\***

```
QDOS MACRO
      MOVEQ £1,D0 TRAP CODE
      TRAP £12 DO IT
      ENDM
```

**FINISH MACRO**

```
QDOS IO_CLOSE,2 Close channel*
MOVEQ £-1,D1
MOVEQ £0,D3 Cancel this Job
QDOS MT_FRJOB,1
MOVEQ £0,D0 Return code - all OK
RTS
ENDM
```

\*Print question

```
QUAN1 MACRO
      MOVEQ £1,D2 length of string
      MOVEQ £-1,D3 infinite delay
      LEA.L 12,A1 pointer to string
```

```
QDOS IO_SSTRG,3 PRINT Question
```

\*Read reply

```
QDOS IO_FBYTE,3 Fetch a byte
MOVE.L D1,D5 Store it on D5
CMPI.B £$2F,D1 Is it < 9
(Suggest enter / to finish)
```

```
BLE FIN Back to ODDS if so
QDOS IO_SBYTE,3 Print the input,
MOVEQ £10,D1 Insert a line so
QDOS IO_SBYTE, send CR
ENDM
```

\* Print Heading or Comment

```
QUAN2 MACRO
      MOVEQ £11,D2 length of string
      MOVEQ £-1,D3 Infinite delay
```

ACK

\*The following 4 lines make it possible to  
\* use values of m & n up to 100 or so, i.e.  
\* enter m,n above 9 as letters A to Z, etc.

```
CMPI.B £$39,D5 Is n > 9 ? i.e., is n
                  A,B,C ...? Thus A is 10,
                  B is 11, Z is 36, etc
BLE PR1 If not, carry on. If it is,
SUBI £7,D5 subtract 7.
PR1 MOVE.L D5,D1 Put n in D1
SUB.L £$30,D1 But not ASCII!!
CMPI.B £$39,D6 Same for m
                  Is m > 9: i.e., is m A,B,C etc?
BLE PR2 If not, carry on
SUBI £7,D6 If it is, subtract 7.
PR2 MOVE.L D6 ,D0 Put m in D0
SUB.L £$30,D0 Again, correct m!
```

**\*\*\* Ackermann\_Function \*\*\***

```
* See Leon Heller QUANTA July page 24
* On entry D0=m, D1=n
* On exit, Ack(m,n) returned in D1
ACK2
TST.L D0 m=0 ?
BNE RECALL NO!
ADD.L £1,D1 YES, return n+1
RTS
```

**RECALL**

```
TST.L D1 m=0, n=0 ??
BNE RERECALL NO !
SUB.L £1,D0 YES, D0=m-1
MOVE.L £1,D1 D1:=1
BSR ACK2 (m-1,1)
RTS
RERECALL
MOVE.L D0,-(SP)
SUB.L £1,D1 D1:=n-1
BSR ACK2 (m,n-1)
MOVE.L (SP)+,D0
SUB.L £1,D0 D0:=n-1
BSR ACK2
```

**RTS**

**\*\*\* BINDEC \*\*\***

\* Converts binary no. to decimal  
\* On entry, low word in D0 holds  
\* no. to be converted  
\* All registers preserved

**BINDEC**

```
MOVEM.L DO-D1,-(SP) Count for 5
digits
```

```

LEA.L  \2,A1          pointer to
*                               string
*                               *
QDOS  IO_SSTRG,3 PRINT Heading etc BINDEC1
ENDM          DIVU    £10,D0
*                               SWAP   D0    Rem in lower word
*                               ADD.B  £'0' D0  Convert to ASCII
*                               MOVE.B  D0,-(SP)  Save char on
*                               *                               stack
MOVEQ    £-1,D1          current job *
MOVEQ    £2,D3          exclusive device
LEA.L   DEVNAME,A0      pointer to DV
QDOS   IO_OPEN,2       Open stream
MOVE.L  £53E000,SP      Give plenty of *
*                               room for stack!
*                               MOVE.W  £4,D2   5 digits on stack
*                               BINDEC2
*                               MOVE.B  (SP)+,D1  Pop char off stack
QUAN2  MS2-MS1,MS1     * 1st Macro *
*                               QDOS   IO_SBYTE,3  Print digit
*                               Prints Headings
*                               DBRA   D2,BINDEC2
AGN
QUAN1  MS3-MS2,MS2     * 2nd Macro *
MOVE.L  D5,D6          Save m on D6
*                               *
*                               Asks for a and save it
QUAN1  MS4-MS3,MS3     * 3rd Macro *
*                               *
*                               Asks for n and has it
*                               Remember that n is on D5
QUAN2  MS5-MS4,MS4     * 4th Macro *
*                               *
*                               Records Ack(m,n)
*                               DEVNAME  DC.W 4
*                               DC.B 'CON_'
*                               MS1     DC.B 10
*                               DC.B 'ACKERMANN FUNCTION'
BSR   ACK              DC.B '          Bill Cowhig '
MOVE.L D1,D0          DC.B 10,10
BSR   BINDEC          DC.B          'Enter / to finish '
JMP   AGN             DC.B          'and return to QDOS'
*                               DC.B 10,10,10
FIN          MS2      DC.B          'Enter Value of m:'
*                               MS3      DC.B          'Enter Value of n:'
FINISH      MS4      DC.B          'Val. of Ack(m,n)'
*                               MS5      DC.B          ' '
*
*
*
END

```

### IQLUG PROGRAM LIBRARY

Cartridges currently available from the sub-librarians (listed below) are as follows :

**LIBRARY 1** : A compendium of BASIC games and utilities (free). There is an option of some machine code procedures and a full screen editor written by Tony Tebby. Send a cheque payable to "QJUMP" for £1 if you want them. The full list of programs appeared in QUANTA issue 8.

**LIBRARY 2** : A compendium of more BASIC games and utilities (free). Includes a 300 baud terminal program, some esoteric statistics programs (thanks Leon), numerous games, a calendar program and much more (33 programs total).

**AMONCPY** : Nothing to do with generating multiple Mr. Andrews's. A cartridge containing three options (a to c), all available from the sub-librarians to whom cartridges and cheques should be sent. Please don't send cash because it makes life more -difficult.

- a. £1 to Mike Finley gets you a copy of his assembler and machine code procedures to speed up the execution of the BASIC it's written in.
- b. £1 to Leon Heller gets a copy of "mon\_exec", an executable machine code monitor for debugging machine code (note that leading zeros have to be supplied and 8 hex digits have to be entered for all addresses).
- c. £1 to Rob Sherratt gets a copy of the "FCOPY" fast cartridge duplicator which is what all the librarians use to rattle off full cartridge copies in 2 to 3 minutes per copy. The program comprises some machine code which does the work and a BASIC driving program which makes everything look pretty.

CASFILE : An extremely well produced and professionally presented set of procedures and screen definitions for use with Archive. Sets up a record and tape library, but could be easily modified for many other purposes. Written by A. Nakhimov and supplied to the group (with author's permission) by Tony Magnus. Includes full instructions, takes up almost the complete cartridge, and is FREE.

### **FUTURE PLANS FOR THE PROGRAM LIBRARY**

The astute among you will realise that no library guide has been published yet for LIBRARY 2 cartridge. It will be included in a separate booklet called "LIBRARY GUIDE" which will be purchasable from Brian Pain for £1.50 per copy and will be obtainable at the end of January 1985. In the LIBRARY GUIDE will be full details of all programs to date and on library cartridges 3, 4, and 5 which (after the end of January) will be obtainable from sub-librarians.

The "CASFILE" set of programs is being modified by Ralph Horton to simplify the job of documenting all the library programs and producing the "LIBRARY GUIDE" document. We intend to keep track of bugs and problems reported on every library program, so please write and let us know of any problems (and preferably corrections) you find. It's too much for just one person to tackle alone.

Some ideas for "useful" programs for the library in case anyone wants to have a go:

Machine Code Extensions to SuperBASIC :

- Read Current Graphics Coordinates
- Sprite Definition and Movement
- Extended Directory Command

SuperBASIC Programs :

- Music / Tune generators
- Graphics programs (like Fancy Font for the BEEB)
- Graphical Adventure games
- 3D maze
- Teaching / educational programs for children

Machine Code Programs (£1 per copy):

- Game of LIFE
- Space Invaders type games
- Lunar Lander with Graphics and sound
- Flight Simulator
- VT100 terminal emulator (the last one has failed to materialise so far)

### **HOW TO ORDER COPIES OF THE LIBRARY CARTRIDGES**

In order to get copies of the IQLUG software you have to write to one of the sub-librarians and do the following :

1. For each library cartridge (or part of cartridge) you want a copy of, please send a separate formatted blank Microdrive cartridge.
2. You need to send the appropriate cheques for any of the optional software you request. Separate cheques are preferred please.
3. Send a letter with your name, address, phone number and an assurance that the programs are for your own use only and will not be redistributed, published or resold without prior authorisation from the IQLUG librarian.
4. Cut out the "library symbol" that appears in the latest issue of QUANTA, and send it with your request for the library cartridge(s) - this will prove that you are a current member of IQLUG.
5. Send your letter and cartridge(s) in a jiffy bag and enclose return postage and a self-addressed sticky label.
6. If you are kind enough to donate programs to the library, then send them to Rob Sherratt. Any programs that involve substantial program skill and effort (normally involving machine code) might be eligible for £1 per copy. Otherwise the incentive in sending programs in is that the more you send in, the faster new cartridges become available.

#### LIST OF IQLUG SUB LIBRARIANS

Ralph Horton,  
The Swan Inn,  
Nr. Braintree,  
ESSEX.

Dennis Croombs,  
27, Lulworth Close,  
Farnborough,  
HANTS.  
GU14 8TP.

Chris Cain,  
CSC Associates,  
32a St Catherine's Road,  
Bitterne Park,  
Southampton.  
SO2 4LS.

Stephen Hewitt,  
Portsdown Vicarage,  
Portsdown Hill Road,  
Portsmouth.  
P06 1BE.

Allen Bertram,  
47 Templegate Crescent,  
LEEDS,  
W. Yorks.  
LS15 OEZ.

Chris Parsons,  
2 Holliers Close,  
Sydenham,  
OXON  
OX9 4NG.

David Beales,  
20 Taylors Avenue,  
Hoddesdon,  
HERTS.  
EN11 8QE.

Brian Davies,  
26 Guy Garth,  
Hedon,  
Humberside.  
HU12 8NP.

Mr. M Grew,  
10 Station Road,  
Ackworth,  
PONTEFRACHT,  
W. Yorks.  
WF7 7NA.

## QL Computing by Ian Sinclair

Ian Sinclair has written many books for the Home Computer sometimes in association with other authors, Dilwyn Jones comes to mind on the Spectrum. I have found the other books that have reached me very helpful. He knows the Home Computer market.

As might be expected his book on the QL is practical and I believe that it will be particularly helpful to those unlucky characters who purchased a QL without experience of other Home Computers and came unstuck on that appalling manual of instruction which hadn't even been proof-read. I'm very much a beginner myself, but had the good fortune to have been in possession of both a ZX81 and a Spectrum and had even used Microdrives, and found grievous problems with the Manual, not least from its lack of an Index, remedied recently of course by IQLUG member Eric Demean. This book has an excellent index.

Ian Sinclair takes a newcomer to computing right through from unpacking. He even has a diagram showing the correct connections for a 13 amp plug, and gives the useful advice that I have rarely seen, of using a two-way coaxial adapter so that the non-monitor user doesn't break the TV aerial socket by giving it the sort of regular wear that the manufacturers had never intended. He also gives advice on tuning the TV and leaves nothing to chance. He makes no assumption of knowledge on the part of the reader. What is most important, he puts nothing into the book that he has not tried out on a real live QL. What is more important still, he ignores the Utilities provided by PSION, apart from using them to show how the Microdrives work, this is needed in order to format the cartridges anyway, and he concentrates on the actual computer language.

I never managed to get my wife to touch either Spectrum or ZX81, but she settled down with this book and what she saw as a 'real keyboard'. Scoffers may scoff but that is the way that she saw it, and believe me, if she begins to show some interest, there's praise indeed.

This book isn't meant to be the definitive text on SuperBASIC, it doesn't pretend to be, but anyone who works his way through every exercise, and I have done, will cope with the vagaries of Sinclair's own manual. Eventually we will all have up-to-date versions of the QL, I've had an AM and now have a JM. Sinclair doesn't say what version he used, but I imagine it was a version preceding the AH. I've never worked out where AH and JM differ anyway. The only snag my wife found was in programs in which there were lines using AT. where the versions I have had needed numbering down first and across second, whereas in this book the numbers were across first and down second. The problem for the absolute newcomer is that it appears to work for small numbers, and it is only when the difference between the number of columns and lines comes into it that problems arise. In a book as valuable to a newcomer to computing as this, it is a very small quibble, and even the rawest of new purchasers will work this out.

Whilst generally very complimentary to the design, he is critical of its foibles, and whilst he gives a number of programs which illustrate the capacity of the QL to produce sounds, he says firmly that the BEEP command is simply adequate. The Beep command may have many variables, but few will deny that it was not really designed with musicians in mind. Sinclair's particular version of the QL, ( No connection with Clive of that ilk) seems to have had problems neither of the versions I've had my hand on have shown. His versions seem to have gone berserk if it had LEFT\$, RIGHT\$, and MID\$ used as COMMANDS. I have tried hard to get mine to respond, they have simply ignored my efforts. Being a simple soul with a SPECTRUM I'd never been used to them anyway. Wherever he meets a command such as CIRCLE R which for him, didn't work, he says so, and accepts that the final version of ROM may have some differences from the one he had on test.

I found his chapters on FURTHER PROCEDURES, WINDOWS, and CHANNELS particularly helpful. The programs themselves are quite short, and many will be useful



in the modular construction of the further fuller programs to which the QL SuperBASIC lends itself so well.

Wil Jackson,  
131 Denton,  
Malden Crescent,  
London NW1

### **Notes on COMPEC, Olympia.**

There was not very much of interest to QL users at the main annual computer industry showcase, COMPEC, held as usual at Olympia.

The only company exhibiting anything for the QL was Quest, who actually had some disk drives hooked up to a QL, which was an improvement on their press launch. Unfortunately, they were not fully operational, and punters were being discouraged from actually using the system.

Logic Programming Associates (LPA) told me they were seriously considering the implementation of PROLOG on the QL. Spectrum PROLOG was done by them for Sinclair. The Hungarian state software organisation are also into PROLOG - they in fact licence LPA as distributors of some of their software - and I got chatting to a lady on their stand who told me that a friend of hers in Hungary had got a QL, and would be interested in joining IQLUG. She told me there were three or four QLs over there. I doubt if the US government would be very happy about this, as they do not like advanced technology getting into Eastern Bloc countries, even in the form of personal computers.

Micro Focus, the CIS COBOL company, said they had no intention of putting their software on the QL, so hard cheese to all you COBOL fans.

I had hoped to see ICL's One-per-Desk, which is based on the QL, but ICL were not showing it off until the official launch in December.

Interestingly, out of about 12 people I spoke to on various stands, four owned QLs, and will probably join IQLUG!

Leon Heller.

### **The ZX-Microfair**

Thanks to Mike Johnstone, we had a free stand at the ZX-Microfair held at Alexandra Palace on the 17/18 November.

Other QL related exhibitors were CST, with their parallel printer interface, GST, with their 68K/OS operating system, Computer One, with their range of QL software, and Adder Publications. Sinclair had a large stand, and seemed to be having trouble getting rid of their stock of QLs, as at first they were giving away a ZX-81 with each QL (someone suggested that this was for use when the QL broke down!), and towards the end of the show they knocked £20 off the price of the QL.

Although CST were not showing it off on their stand, they actually had one of their disk interfaces with them, which is now in production form, with a proper PCB and plastic case to match the QL. There is quite a lot of spare PCB area that they told me would be used for an optional 128K RAM expansion. The official launch of the unit will be in December, and the interface will cost £250 with two 3.5" drives, or £100 on its own, for use with your own 3.5" or 5.25" drives.

We managed to persuade about 70 users to join IQLUG during the course of the show, so it was quite worth the effort involved. My thanks to the members who helped out on the stand.

Leon Heller

### SHINWA CP-80, type 1 printer

With the help of Sinclair Research's Customer Relations officer (not Psion), I have succeeded in installing the above printer. As other members may be having the same problems I had, I set out below the relevant codes.

I am using the 'QL Centronics Printer Interface', made by 'Miracle Systems Limited.'

	<u>ASCII</u>	<u>Decimal</u>
Driver name	: Shinwa CP80	
Parity	: none	
Baud rate	: 9600	
Lines/page	: 66	
Characters/line	: 80	
Continuous forms	: Yes/no	
End of line code	: <CR>,<LF>	
End of page code	: <FF>	
Preamble code	: <ESC>,'@'	27,64
Postamble code	: none	
Emphasis. On	: <ESC>,'E'	27,69
Emphasise off	: <ESC>,'F'	27,70
Underline on	: <ESC>,'1',1	27,45,49
Underline off	: <ESC>,'-',<NUL>	27,45,00
Subscript on	: <ESC>,'S',1	27,83,49
Subscript off	: <ESC>,'T',<ESC>,'H'	27,84,27,72
Superscript on	: <ESC>,'S',<NUL>	27,83,00
Superscript off	: <ESC>,'T',<ESC>,'H'	27,83,27,72
Translate 1		96,129

Three of the above are not given by either the QL or the Shinwa manual: underline off, superscript on and translate. The '00' given under 'decimal' for the underline off and the superscript on is really Hex - Quill does not respond to the decimal zero (48) or the ASCII zero ('0').

Neither underline off nor superscript on will work if the codes indicated in the manuals are used. I have tested the Shinwa codes using Basic programmes as illustrated in their manual and they work properly. Clearly, the problem is in Quill, and Psion indicate that QLUB members will receive an update version of Install\_bas 'shortly'. Is this good news now that I have been supplied a solution ?

Using Translate 1, the keyboard '£' sign is converted from what the Shinwa interprets as a single opening inverted comma to the £. (I cannot print the ASCII equivalent as I have converted it!). The Epson does not require the "<ESC>,'H'" code on superscripts or subscripts: the Shinwa needs them to turn off the double-print mode which is automatically invoked by these two features.

I hope other members will find this useful.

Roy Ashworth,  
55 Half Moon Lane,  
London SE24 9JX.

### EASEL graph printing to Brother HR-5 printer

First press and release the RESET button.  
Place the EASEL cartridge in MDV1 and type:  
Let a=RESPR(3000)  
LBYTES mdv1\_gprint\_prt,a  
Poke\_w a+94,3000

```
delete mdv1_gprintprt
sbytes mdv1_gprintprt,a,2000
```

```
Now type: load mdv1_boot
list (lists EASEL boot)
10 baud 9600
20 open #3,Ser1
30 print #3,chr$(27);'1'
40 print #3,chr$(27);'U';chr $(1)
50 print #3,chr$(27);'K';chr$(224);chr$(1)
    (these lines added to EASEL boot)
delete mdv1_boot
save mdv1_boot
    (changes new boot for old)
```

HR-5 DIP switch settings. Note that the two arrays of switches are numbered from front to back; the following settings correspond to this arrangement:

SW 1-8 OFF	SW 2-6 OFF
1-7 OFF	2-5 OFF
1-6 OFF	2-4 OFF
1-5 ON	2-3 OFF
1-4 ON	2-2 ON
1-3 OFF	2-1 ON
1-2 ON	
1-1 OFF	

QUILL Install\_bas :-

Driver name	:	HP-5	
Parity	:	None	
Baud rate	:	9600	
Lines/page	:	6	
Characters/line	:	80	
Continuous forms	:	Yes	
End of line code	:	CR,LF	
End of page code	:	FF	
Preamble code	:	Esc,2	: 27,2
Postamble code.	:	Esc,1	: 27,1
Emphasise on	:	Esc,E	: 27,101
Emphasise off	:	Esc,F	: 27,102
Underline on	:	Esc,-,SOH	: 27,45,1
Underline off	:	Esc,-,NUL	: 27,45,0
Subscript on	:	Esc,S,SOH	: 27,115,1
Subscript off	:	Esc,T	: 27,116
Superscript on	:	Esc,S,NUL	: 27,115,0
Superscript off	:	Esc,T	: 27,116
Translate 1	:	£,£	
Translate 2	:	None	

(Type in the codes in the right hand column to get the display in the centre column. Otherwise enter the commands as they stand\_ Editing the FX-80 list makes it simpler.)

DR. BASIL LEE,  
31 FAIRDALE GARDENS,  
LONDON. SW15 6JW.  
Tel: 01-789 1976.

SPECTRUM AS SPOOLER FOR QL

61A8           ORG     25000  
61AB           ENT     25000

:INITIALISATION

61A8 3E01       LD     A.01       ;       network device nr. 1  
61AA 32D65C    LD     (23766),A  
61AD CF        RST     8        ;       open network  
61AE 2D        DEFB    '.'  
61AF 219C63    LD     HL,25500    ;       buffer starts at 25500  
61B2 22F161    LD     (IN).HL     ;       set in-value to 25500  
61B5 22F361    LD     (OUT).HL    ;       set out-value to 25500  
61B8 DD22515C  LD     (23633).IX

:This is the input routine

61BC CDE615     INPUT CALL #15E6   ;       input byte from network  
61BF 3008       JR     NC,READY   ;       no:test: is printer ready?  
61C1 2AF161    LD     HL,(IN)     ;       yes,get in-value  
61CA 77        LD     (HL),A     ;       store byte  
61C5 23        INC     HL        ;       increment in-value  
61C6 22F161    LD     (IN).HL     ;       store new in-value

:test: is printer ready?

:if printer ready, test for token to be printed

61C9 DBEF   READY IN   A.(#EF)   ;       test port \$ EF  
61CB CB5F            BIT   3,A     ;       is DTR Zero?  
61CD 28ED            JR    2,INPUT ;       yes:go to INPUT  
61CF 2AF161    TOKEN LD   HL,(IN) ;       get in-value in HL  
61D2 ED5BF361   LD   DE,(OUT);       get out-value in DE  
61D6 A7            AND   A       ;       reset flag  
61D7 ED52           SBC  HL,DE   ;       in - out - zero?  
61D9 28E1           JR    Z,INPUT ;       yes:go to INPUT

: now a token is controlled and after necessary modification :sent to RS 232 -  
: you can add here modifications according to your printer

61DB 2AF361           LD   H1,(OUT);       get out-value  
61DE 7E            LD   A ,(HL) ;       get token to be printed  
61DF FEOA           CP   #0A   ;       Is it #0A ?  
61E1 280A           JR   Z,CR   ;       yes: go to change routine  
61E3 E5            PRINT PUSH HL  
61E4 CF            RST #08   ;       token is sent to RS 232  
61E5 1E            DEFB #1E  
61E6 E1            POP HL  
61E7 23            INC   HL     ;       Increment out-value  
61E8 22F361       LD   (OUT).HL;       store new out-value  
61EB 18CF           JR   INPUT ;       go to INPUT

:change routine

61ED 3E0D       CR     LD   A,#0D   ;       load A with changed token  
61EF 18F2        JR   PRINT   ;       go to PRINT  
61F1            IN     DEFS 2   ;       In value is stored here  
61F3            OUT    DEFS 2   ;       Out-value is stored here  
61F5            END

## THE TAB FUNCTION ON THE QL.

I HAVE RECENTLY REQUIRED A TAB FUNCTION TO PRINTER IN A PROGRAMME I WAS DEVELOPING AND ENQUIRIES FROM IDUG MEMBERS YIELDED SOLUTIONS WHICH WERE SATISFACTORY BUT SOMEWHAT UNWIELDY IN MY PARTICULAR APPLICATION. MY THANKS TO BRIAN PAIN AND ROB SHERRATT FOR THEIR ASSISTANCE.

HOWEVER I HAVE FOUND THAT A TAB FUNCTION IS IMPLEMENTED IN THE QL, IT GOES LIKE THIS:-

PRINT £6. TO 20:" This is position 20"

THIS INFORMATION IS PRINTED IN THE NEW VERSION OF THE MANUAL ISSUED WITH NEW QL'S FROM HIGH STREET OUTLETS. HOW MANY MORE ITEMS ARE IMPLEMENTED BUT NOT SHOWN IN THE LITERATURE? A COMPARISON BETWEEN OLD AND NEW VERSIONS OF THE MANUAL MIGHT BE WORTHWHILE.

Mr J.R.Haldane,  
16 Burnside Road,  
Gorebridge, Midlothian. Scotland.

## RECOVERING CRASHED QUILL DOCUMENTS

I decided it was time to make some contribution, however humble, to your worthy organ, which has become compulsive reading in this Department even though my QL is the only one hereabouts that I know of.

Readers may be interested in a rather crude but effective method of recovering crashed Quill Documents (version 1.03 ACB).

1. Remove Microdrive cartridges. If drives "gwine to run all night", skip this step.
2. Press Reset. Replace document Microdrive only. Select F1 or F2.
3. Copy def\_doc to some suitably-named file with a \_doc suffix, e.g. <Copy mdv2\_def\_doc to mdv2\_gash\_doc>.
4. Re-boot Quill. DO NOT attempt to load gash\_doc or Quill will throw in its hand. Instead, use the print command to write out the renamed file to another suitably named file with suffix lis, e.g. gash\_lis (Quill appends the suffix automatically).
5. Finally, use the Files command, Import option to read in gash\_lis. Set the Design, Header and Footer as required, save it to the name required and use Delete to get rid of "gash\_doc" and "gash\_lis". Then carry on normally.

JS Medhurst,  
School of Marine Technology,  
Armstrong Building,  
The University,  
Newcastle upon Tyne.  
NE1 7RU.

## QL CARRYING CASES

I have finally found some carrying cases for the QL. They are made by the following company : Praybourne Ltd., Unit 5, Millsborough House, Ipsley Street, Redditch, Worcs. B98 7BU. Telephone: 0527-61221. The soft case is 0.5" larger in all three dimensions than the QL and its transformer, approximately 22"x 2.5"x 6". It weighs about 1.5 pounds

(0.75 kg.) It costs £17.50+VAT. £1.95 P/P = £22.08. Allow about 1 week for delivery from the time they have received your cheque and order. They have sold over 350 already. The soft case is made of a nylon coated P.V.C. material with a high impact 0.25" thick foam lining and a pouch on the outside.

The company also does a fibre box case for the QL which weighs over 3.1 Kgs. and costs about £30. They also say they haven't sold one of those yet.

Donald H Gehring

### **Hardware Review : PRISM QL 14 COLOUR MONITOR By Roy A. Woodward.**

After using the Sinclair QL for many weeks with a domestic 14" TV, and suffering the eyestrain-inducing display, I eventually decided it was time to look for a suitable monitor at a reasonable price. While appreciating the improvement a monitor should bring, I could not justify spending 300 or more pounds on a top flight RGB monitor such as a Microvitec, for the £400 QL computer.

The lack of display quality using any micro with a domestic television is generally a result of two things:

Lack of resolution impairing the quality of say an 80 column display; and Dot crawl or shimmering on coloured areas of the display.

Both of these problems are partly caused by the way the computer produces the TV display : The video signal in the computer is modulated onto an RF carrier wave, so when it is sent down the cable to the TV it appears as if it were a signal being transmitted across the airwaves. The TV then has to demodulate the signal to arrive once again at suitable video levels to produce the display. It is this modulation and demodulation which causes a loss of quality and lack of resolution. The dot crawl problem is, I suspect, aggravated by cost cutting at the design stage, and poor setting up of each machine, as some QLs show more dot crawl than others.

So, having analysed the inadequacies of the RF output, let us look at the Prism QL 14. This monitor is marketed by Prism, one of the major Sinclair distributors, and is badge-engineered for them by Fidelity Electronics. The general style of the casing bears a striking resemblance to the Fidelity CTM 1400, although the QL 14 is purely an RGB monitor, with no TV tuner and no sound output.

The QL 14 monitor is available for £199.95, and I obtained mine at my local Co-Op computer department. The unit comes sturdily packed in a large cardboard box, complete with instruction book and a suitable cable for the QL. This has an 8 pin DIN plug for the QL, and terminates in a 21 pin Euroconnector which plugs into the rear of the monitor. The use of this standard 21 pin plug makes me suspect that the QL 14 monitor may be compatible with other computers given the correct lead.

The monitor is finished in QL matt black, and has a flat smoked glass filter fitted in front of the tube. Thoughtfully, this filter can be removed easily to clean behind it, although this is not mentioned in the manual. The overall visual impression is that of a smart, functional unit. The monitor has only two controls, an on/off pushbutton at bottom front (with an LED power indicator just above), and a contrast control behind the hinged front panel. Apart from these, the QL 14 is all internally preset to work with the Sinclair QL, literally plug in and go.

So what about display quality? The QL 14 gives a very stable display with good solid colours - no shimmer or dot crawl - and while it is not up to the quality of the Microvitec QL model, it has no trouble in resolving the QL's 85 column output. Using QL Quill on the monitor is a joy - no more eyestrain or headaches, and QL Easel can at last produce some really stunning displays. The monitor copes well with the QL's paper

stipples, giving very stable pseudo colours, and unlike any standard RGB monitor the QL 14 does not lose a single character from either side of the display.

In conclusion, the Prism QL 14 offers a relatively low cost way to improve the QL output. I would recommend it.

### Settings for the Brother EP 44 Printer

I enclose details of how to set-up a Brother EP 44 printer. You may publish it in QUANTA, keep it as information, or throw it in the bin, as you wish. The method I use allows the Brother to be used by QUILL and listings/printouts, although to obtain the '£' symbol in a listing a line-by-line copy must be used, otherwise both the '#' and £ symbols can be printed. Also underlining works well but **bold** does not make any noticeable difference, it just slows the printing.

I have found that it is not a good idea to remove the TAB stops at the end of a line, as this causes the printer to skip pages between lines.

```
100 REMark LIST Kevin Bean 10/11/84
110 CLS#0:CLS
120 PRINT"LIST":PRINT"====":PRINT
130 PRINT"This Program Lists Programs to a printer. Characters with different ASCII
codes than the QL are swapped. " : PRINT
140 PRINT "Line 240 contains the swapping routine.":PRINT
150 PRINT"This version was written for the Brother EP 44, but may need some
adaptations for different printers":PRINT
160 INPUT"Enter File Name ? ";f$
170 CLS:PRINT"LISTING ";f$
180 OPEN _IN#4,f$
190 OPEN#3,ser1c:BAUD 300
200 REPEAT get
210   IF EOF(#4) THEN EXIT get
220   INPUT#4,a$
230   FOR I = 1 TO LEN (a$)
240     IF CODE(a$(i)) =96 THEN a$(i) = CHR$(156) :REMark change pound sign
250     NEXT i
260   PRINT#3, a$
270 END REPEAT get
280 CLOSE#3:CLOSE#4
290 PRINT "LISTING FINISHED"
```

```
BAUD RATE       : 300
BIT LENGTH      : 8
PARITY          : N
NEW LINE        : CR
CODE            : 8BIT
ER              : Y
```

### Settings for the PSION printer driver

```
DRIVER NAME     : BR EP44
PARITY          : NONE
BAUD RATE       : 300
LINES/PAGE      : 66
CHARACTERS/LINE : 80
CONTINUOUS FORMS : YES
END OF LINE CODE : <CR>
END OF PAGE CODE : <FF>
PREAMBLE CODE   : NONE
```

```
POSTAMBLE CODE      : NONE
EMPHASIZE ON        : NONE
EMPHASIZE OFF       : NONE
UNDERLINE ON        : DEF
UNDERLINE OFF       : DEF
SUBSCRIPT ON        : NONE
SUBSCRIPT OFF       : NONE
SUPERSCRIPT ON      : NONE
SUPERSCRIPT OFF     : NONE
TRANSLATE 1         : £,156
TRANSLATE 2         : @,@ (Change 'copyright' to '@')
```

Kevin J. Bean,  
XTD/11 PDO,  
PO Box 81,  
Muscat,  
Sultanate of OMAN.

### RECURSIVE FUNCTIONS ON THE QL

An elegant feature of SuperBASIC is that functions may be defined recursively; that is, you tell the computer how the value of the function relates to other values of the function, and the computer will do the rest for itself.

For example, the mathematical function "factorial", defined as

$$n! = n*(n-1)*(n-2)*...*3*2*1,$$

can be defined recursively as

$$n! = n*(n-1)!$$

(in words: factorial n is equal to n times factorial n-1). In this form it is quite suitable for giving as an instruction to the QL, provided you give it also a starting value, say  $1! = 1$ , so that the computer knows when to stop.

A short and simple program is then sufficient to get the QL to calculate factorials for you, viz:

```
100 INPUT number
110 PRINT factorial(number)
120 DEFine FuNction factorial(n)
130 IF n = 1 : RETurn 1
140 RETurn n*factorial(n-1)
150 END DEFine
```

(This is the sort of thing the QL User Guide means when it says, in its inimitably lucid fashion, "Function calls in SuperBASIC can be recursive, that is, a function may call itself directly or indirectly via a sequence of other calls" (Keywords section, p.15).)

However, in some cases, using recursive definitions can lead to problems. G.N.Howard pointed this out in the October issue of Quanta (p.24), where he showed that an apparently harmless little recursive definition can lead you into unmanageably large numbers before you know what's hit you. But there can also be a problem, even where the numbers are simple and manageable, arising from the length of time it takes the computer to carry out its side of the recursive calculation.

For example there is a sequence of numbers, well known to lovers of rabbits and pineapples, called the Fibonacci numbers. Here each number is defined as the sum of the preceding two numbers, so the sequence goes:-

1, 2, 3, 5, 8, 13, 21, 34, ... & so on.

This is an obvious candidate for getting the QL to use its recursive abilities to churn out as many of these numbers as you may want. The program is straightforward, analogous to the factorial program above:



```
100 INPUT number
110 PRINT Fibonacci(number)
120 DEFine FuNction Fibonacci(n)
130   IF n = 1 : RETurn 1
140   IF n = 2 : RETurn 1
150   RETurn fibonacci(n-1) + fibonacci(n-2)
160 END DEFine
```

This program will give you any particular Fibonacci number. when you input its location in the sequence; so Fibonacci(4) = 3, Fibonacci(9) = 34, Fibonacci(31) = 1,346,269; & so on.

All fine & dandy, except that the QL takes rather a long time to return this last number: to be precise, 8 hours, 40 mins, 41 secs. This is something of a surprise to those of us accustomed to considering computational time of no consequence with the trendy high-tech microcomputer of today. Evidently recursion can be quite a time-consuming operation! In the time it takes the QL to compute one of these numbers, you can have worked it out by hand five hundred times over; or can have calculated by hand, or abacus, a Fibonacci number many magnitudes larger.

The time cited above for the calculation of the 31st Fibonacci number is not an arbitrary number - after all, the QL is a precision machine -but is in fact the sum of the time taken to calculate the preceding two numbers in the sequence: 5 hrs, 21 mins, 48 secs; and 3 hrs, 18 mins, 53 secs. That is, to produce the numbers 832,040 and 514,229 respectively. In other words, the length of time the QL takes to compute successive terms in the Fibonacci sequence is itself a Fibonacci sequence.

Why this should be is not difficult to see, by examining line 150 of the program: you are telling the computer to evaluate two things and Add them together, so the length of time it takes is the sum of the times taken over each evaluation.

Nevertheless, although it may be 'explained away', this observation is an interesting example of the relationship between what you want the computer to do for you and the way it does it.

That aside, the moral of all this is that real-time practicalities should be considered before using the QL's recursive function facilities.

John Fauvel,  
5 Marshworth,  
Milton Keynes. MK6 3DA.

### ARCHIVE LIMITATIONS

I have had my QL, version AB, some 2½ months now. During the period waiting for it, I prepared my 'den', installing power points, RF Filter etc., and a cooling fan. We decided to install an hour meter and this is fitted so that the hours indicated show the number of hours the computer has been 'in use'. We have done 398 hours so far. The main purpose for the purchase was to store my genealogical data, and to use the Archive data base software to implement this. I had developed a program for analysis of the 1851 Census data on the BBC micro as well as my department's PDP11-80 and I expected to write it in BASIC/SuperBASIC on the QL. Many of the features of the SuperBASIC were incorporated at once, but others left until better understood. My background in computing is as a Maintenance Engineer for some 20 years now, entirely on mainframes, never venturing into programming in detail until 2 years ago.

Several QL problems appeared quite quickly - QUILL would not load on my machine or that of a colleague's - 'bad line' etc. It was returned to Sinclair and replaced 28 days later - same problem - and on my friend's JM version. We 'cloned' his version for me and that is OK on my machine. The third QUILL received last week - only 11 days this time - will also not load and cannot be copied from either Microdrive.

Archive gave no problems until my number of records became quite large. All my testing was done on a dummy file of only 100 records with 14 fields. I was able to use all functions - SELECT, SEARCH etc., and ORDER on up to 4 fields. The problem came with real data and my file of 708 records. I had studied the manual, accepted that there would be some limitations despite what it says - assumed that I could only use one cartridge per file - it doesn't say anywhere that you can spread a file over onto another cartridge, but it specifically says that a file is not limited by memory (page 54, para. 6, Error 80 details). This is not so; after much hair-tearing I wrote to PSION. Their reply by return (no criticisms here) confirmed my fears - there are restrictions. It is a pity that the manual doesn't advise us of them. Briefly, of the 12K memory available, 8K is allocated to an index that stores "certain information about each record".

The overheads are:

6 bytes per record with no ordered fields  
14 bytes per record with one ordered field  
22 bytes per record with two ordered fields  
30 bytes per record with three ordered fields  
38 bytes per record with four ordered fields.

To calculate the number of records that you can store in a file, determine how many fields you will need to order, if any, and read the number of bytes from the table, divide this into 8192. This will allow the following -

No order = 1365 records  
1 order = 585 records  
2 order = 372 records  
3 order = 273 records  
4 order = 215 records

Some penalty! The reply goes on to say that procedures grab this area also - so use a "start-proc" Archive, page 29, and chain your procedures. It is more professional anyway. SCREEN also takes some - so beware!

My fix is to SELECT a portion of the file, check how many records with COUNT() and manipulate as required. It is a nuisance, but is a way round the restriction. Incidentally, when the message "out of memory" comes up, it is not always possible to recover the situation using Break or CTRL spacebar, a reset may be necessary. This of course loses all open files - more cursing. ALWAYS copy your data and programs frequently - DON'T use Backup to the same cartridge - a damaged tape and you cannot get at your file-bak. After Data entry I CLOSE all files and then "look" - a crash in the system will not lose your data. The cost of extra cartridges is very soon cost effective if you find you have to enter the data again - I did - it took 20 hours. Now I copy everything to other cartridges, get up and have a coffee, etc., everything is completed on your return. Use MEMORY() and COUNT() often - or use a procedure to check the area of memory left - this too will save your temper; a pity that this test is not available in SuperBASIC as a keyword e.g. HIMEM/LOWMEM.

PSION infer that a version 2 is on the way - I hope that it will be clear of some of these bugs.

M F Jackson, Hollington House, 74 Woodfield Road, Cam, Dursley. Glos. GL11 6HF.

### ZERO ELECTRONICS PRINT BUFFER

Faced with my new AL without a parallel interface, and my trusty Seikosha GP80A printer, I had to decide what to do for the best. Ideally I would have liked to replace the Seikosha with an Epson with serial interface, but finances prevent that at present. So should I get a converter, and in due course a standard Epson? Why not make the 'black box' between the computer and printer do more than just convert from serial to parallel?

The solution was a serial input/parallel output printer buffer from Zero Electronics of 149 King Street (Nash House), Great Yarmouth.

To cut a long story short, their 16K unit cost me £110 including VAT and carriage. Setting the appropriate dip switches inside the unit started off well by referring to the QL and Zero manuals, but the last few settings were made by trial and error. The difference even 16K of buffer makes to most computer applications is incredible. It is almost as good as having true multi-tasking! You can send a document to be printed from Quill, and have time to start loading another package before the printer has finished the buffer contents.

I have been able to borrow an Epson for long enough to confirm that the same settings will work. I also found that the Psion Epson FX80 printer driver seems compatible with the earlier MX100. The advantage of a buffer over a spooling program is that it can be used from within the packages without the need to write your file to a Microdrive first. Reports from Archive are speeded up considerably, as the program does not have to wait for the printer at all, and a procedure may finish and control be regained long before the hard copy has all been printed.

To save anyone else who might get a Zero buffer some time, these are the settings which I am using for 9600 Baud.

d	c	b	a	1	2	3	4	5	6	7	8
on	on	on	on	on	off	off	off	on	on	off	off

I have had to use Applewriter II and an Epson MX100 to write this! Does anyone want a Seikosha GP80A and Video Genie Interface for £100, or an Aculab Floppy Tape Unit for a Video Genie, plus XBAS and SCRIP for £70?

Pete Fisher,  
2 Belvedere Avenue,  
Penn,  
Wolverhampton.  
Tel 0902 34133

### **MICROLINE 82A PRINTER**

I have recently bought a Microline 82A printer from Mancos Computers of Manchester for £245.45 including VAT and carriage.

The carriage was so fast as to be unbelievable - the printer arrived within 24 hours of posting the order!

Installing it was not quite as easy - mostly due to ignorance on my part, of course.

Only three wires are used from the QL - Signal Ground (pin 1 BLACK), Data Transmit (pin 2 WHITE) and Handshake (pin 4 BLUE); these go to pins 7,3 and 11 on the D plug and according to the Users Manual pins 20 and 6 should be connected but it doesn't seem to make any difference. The plug should go to Ser1 on the QL.

Whilst trying to measure the polarity of the Handshake signal with a voltmeter connected across pins 7 and 11 the printer worked O.K. but on disconnecting the meter spurious graphics reappeared.

This fault was cleared by replacing the meter with a small value capacitor (.005 to .01 microfarad) - a resistance had no effect.

There is a DIP switch on the main circuit board which sets the polarity of the Handshake - this requires switch 1 to be ON. Switches 2,3 and 4 set the the baud rate (I have used 1200 - the top speed for the interface provided in the printer).

The advertised print speed is 120 cps in fact it seems to go at about 80 cps. However it does provide a reasonable printer at reasonable cost.

If anyone would like to talk to me about this printer my phone number is Great Yarmouth 665984

Ralph Thomas

## PASCAL REVIEW

For the past month now, I have had a copy of Computer One's PASCAL in my hands. Having started with version 1.00, which worked on AH QLs but not on my JM QL, v1.01 came and still had a few small bugs in, then finally v1.02 arrived.

For a modest outlay of £39 you get a Microdrive cartridge and a 90 page spiral bound book. The cartridge contains a boot program and a clone program as do the PSION packages as well as the compiler, a full screen editor and 4 demonstration programs. On booting you are presented with an 8 option menu, offering the usual range of edit, compile, run, copy, dir etc. One nice function in that you can return to SuperBASIC, but code is left in the resident procedure space, so typing 'Pascal' boots you back into Pascal.

The editor takes 15 secs to load, and presents you with another menu, of 7 options. The editor itself is easy to use and refreshingly fast after the agony of using Quill. It works in pages so time is saved from continually scrolling the screen, which is an advantage with the QL's very slow screen i/o. To skip through a file fast, you can page it or fast move is provided with an indicator arrow which moves down the side relative to the length of the file. Auto-indentation is provided so the return starts directly below the previous line, and is not as useful as the intelligent indentation of Archive's editor. The editor doesn't possess a command mode and so lacks the power of editors such as TXED and ECCE, although it does have a simple and very quick string search function. My one real grumble with the editor is that there is no facility for either moving text or merging text from another file, although Computer One are working on this at the moment. This is a real disadvantage with a language like Pascal which being in modular form lends itself to the development of software tools. This can be replaced by writing a library linker, such as the one I have given at the end of this review. This is a filter which copies its standard input to its standard output but replaces the statement ( # include frame lib) by the contents of frame. Future editions of Quanta will contain more Pascal tools. On balance, the editor is certainly an improvement on Quill and the SuperBASIC editor for writing BASIC programs!

On to compilation ... The compiler takes a minimum of 45 secs to do its job; this is slow if you are used to Turbo Pascal, but a good proportion of the blame should be placed on the Microdrives for this. The compiler loads separately, and produces a compact intermediate p-code (average half the length of the original Pascal program) which is interpreted at run time by a resident interpreter. Computer One originally wanted to produce a compiler that went straight to 68000 code but thought it wasn't feasible with the memory limitations of the QL: this means that Pascal programs tend to run around 21/2 times faster than their SuperBASIC equivalents rather than the 5 times one might expect from native code. I would like to see this system running with a 128k memory add-on configured as a silicon disk to save having to keep loading the compiler and editor; it would make debugging programs much more pleasant. If any errors are found, the compiler produces a file frame\_err which the editor takes and when you reload the file it inserts the error messages in the appropriate place. However, empty lines at the head of the file can dislocate the error messages from the line they ought to be associated with, causing a feeling of dyslexia in the user!

The Pascal itself is fairly close to standard Jensen and Wirth Pascal, though with Pascal purity of language is hardly an asset. The non-standard stuff is mostly concerned with i/o, which the original Pascal definition contains notoriously little of. The only

regrettable loss is the ability to pass functions as parameters. Computer One have provided the ability to use almost all of SuperBASIC's built-in functions and procedures (probably on the grounds that the kind of person who buys a Sinclair computer isn't interested in portability anyway!)

In addition they have provided several new functions which I feel provide one of the most interesting parts of this package. These include getline and putbytes procedures for easier file handling, which could be used in place of my readline and putline procedures in the library linker. They also provide a function which returns the location of a variable. This means that lines such as filenames can be read in interactively, and allow the use of the editing keys in the process, which would be relatively complicated to do from standard Pascal. The other really interesting additions are concerned with allowing machine code sections to be called easily from Pascal. They have provided predeclared arrays for the data registers and the address registers of the 68008 which are passed automatically on calling a machine code subroutine, which is ironic since the original definition of Pascal deliberately forbade access to machine code and the CPU from Pascal. However, the elegance of these can be seen from one of their demonstration programs. This is a clone program with a procedure called trap, which is well worth reading in conjunction with Adrian Dickens Advanced User Guide. Another predeclared variable ioresult is also returned after i/o allowing the user to decide whether to halt a program or not if there is an error - the use of this is illustrated by my version of the function 'mustopen'. This compiler does not go as far as Turbo Pascal has done in implementing BASIC-like string handling. It is interesting to watch Pascal evolve towards BASIC in a similar though less pronounced way than BASIC has towards the structured aspects of Pascal.

My only grumbles are that the interpreter uses the QL's own maths routines with their few significant digits (MSX offers 14 in double-precision mode to a maximum of 8 on the QL) - this is a shame since part of the original idea behind Pascal was the structuring of data as well as the structuring of programs - and that there is no break function if the program goes into an endless loop. Read will not allow real numbers starting 1.1, they cause the program to crash, so be warned.

On the whole then this is a very straightforward implementation of Pascal offering full access to all the QL's capabilities, and represents excellent value for money, though one wonders how it will compare with other Pascal compilers that will be around in 6 months or so.

program liblink ; (replaces [# include fnamelib] by the contents of fname)

```
const      newline = 10 ;
           Maxstr   = 96 ;
           Namlen   = 17 ;
           lread    = 1 ;
           lowrite  = 2 ;
           lf       = "n" ;

type fline = array[1..maxstr] of char ;
           fname = array[1..namlen] of char ;

var        for,
           Inlen : integer;
           inane,
           oname : fname ;
           Ofile : text ;
```

procedure readline (var source : text ; var s : fline) ;

```
var i : integer ;
begin
  i := 1 ;
  while (not eoln (source)) and (i < maxstr) do
    begin
      s[i] := source^;
      i := succ(i) ;
```



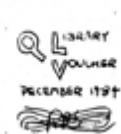
```
    get (source) ;  
end ;  
s[i] := chr(newline) ;  
get (source) ;  
end ;
```

```
procedure putline(var gaol : text ; var g : fline) ;  
var i : integer ;  
begin  
  i := 1 ;  
  while g[i] <> chr(newline) do  
    begin  
      gaol^:= g[i] ;  
      i := succ(i) ;  
      put (gaol) ;  
    end ;  
  gaol^:= chr(newline) ;  
  put (gaol) ;  
end ;
```

```
function finclude (var s : fline) : integer ;  
var l, ret : integer ;  
begin  
  l := 1 ;  
  ret := 0 ;  
  while s[l] = ' ' do l := succ(1) ;  
  if s[l] = '£' then  
    begin  
      i := succ(i) ;  
      while s[i] = ' ' do i := succ(i) ;  
      if s[i] = '#' then  
        begin  
          i := succ(i) ;  
          while s[i] = ' ' do l := succ(i) ;  
          if (s[i] = 'l') and  
             (s[i + 1] = 'n') and  
             (s[i + 2] = 'c') and  
             (s[i + 3] = 'l') then ret := i + 7 ;  
        end ;  
      end ;  
    finclude := ret ;  
  end ;
```

```
function mustopen  
  (var f : text ; var n : fname ; mode : integer) : integer ;  
var iok : integer ;  
begin  
  attach(f, n) ;  
  if mode = ioread then  
    begin  
      reset (f) ;  
      iok := ioresult  
    end
```

```
  Else  
    if mode = iowrite then  
      Begin  
        rewrite (f) ;  
        iok = ioresult  
      end  
    else iok := 1 ;
```



```
if iok <> 0 then
  writeln (lf, ' I/O error in mode: ', mode:2) ;
  austopen := iok ;
end ;
```

```
procedure insert (var outfile : text ; inname : fname) ;
var i, j,
    lor : integer ;
    inline : fline ;
    libnam : fname ;
    infile : text ;
begin
  if mustopen (infile, inname, ioread) <> 0 then
    writeln (' File not copied ')
  else
    begin
      while not eof (infile ) do
        begin
          readline (infile, inline) ;
          i := finclude (inline) ;
          if i > 0 then
            begin
              j := 5 ;
              libnam := 'mdv2' ;
              while inline[i] = ' ' do i := succ (i) ;
              repeat
                j := succ (j) ;
                libnam[j] := inline[j] ;
                i := succ (i) ;
              until (libnam[j] = ' ') or (j - namlen) ;
              writeln (' ' n Library module: ', Libnam, ' ' n') ;
              writeln (outfile) ;
              insert (outfile, libnam) ;
              writeln (outfile) ;
            end
          else
            begin
              write ('* ') ;
              putline (output, inline) ;
              putline (outfile, inline)
            end ;
          end ;
        end ;
      close (infile) ;
    end ;
  end ;
end ;
```

```
begin
  border (output, 18, green, green, 0) ;
  csize (output, 2, 1) ;
  writeln ' PASCAL library linker ) ;
  csize (output, 0, 0) ;

  write (' ' n Name of source file: ' ) ;
  inlen := getline (input, namlen, loc(inname));
  if iname[inlen] = lf then iname[inlen] := ' ' ;

  write (' Name of output file: ' ) ;
  inlen := getline (input, namlen, loc(oname));
  if oname[inlen] = lf then oname[inlen] := ' ' ;
```

```
if mustopen (ofile, oname, iowrite) <> 0 then halt ;
insert (ofile, iname) ;
close (ofile) ;
```

end .

Roger MacNicol

### USING SERIAL INTERFACE TO CONNECT BBC TO QL.

The SINCLAIR QL has a second serial interface (ser2) with lines crossed, relative to ser 1, so that there is no need to use a turn-round connector. However if using the RS 232 cable supplied, note that it will be necessary to cross connect QL pin 20 to BBC pin 4. (The QL manual is slightly misleading on this point. "Concepts" page 12 says "ready input" is on pin 4 - as required - but this is pin 4 of the D-plug, which goes to pin 20 of the RS 232. Pins 2, 3, 5 and 6 of D-plug all go to the same-numbered pin of the RS 232). It might therefore be worth making up your own cable with D-plug at one end and BBC DIN plug at the other.

One of the differences between the BBC and the QL is that programme lines on the BBC are separated by <CR><LF> (both are supplied automatically by the operating system on pressing RETURN). On the QL only <LF> is supplied on pressing ENTER. Although this makes for difficulties obtaining hardcopy of QL programmes, it is not a problem when transferring between BBC to QL: using ser2c, each computer accepts as input the output of the other.

Another difference is that the BBC default is 8 bits + 1 stop bit, no parity; the QL default is 8 bits + 2 stop bits, no parity. However both default to 9600 baud.

So the following can be used to transfer from BBC disc storage to QL:

```
on the BBC:  *FX 5,2
              *FX 156,82,0
              LOAD "program"
              control-B
              List
```

```
on the QL:   load ser2c
```

(nothing is displayed on the QL screen, though the BBC screen displays normally). At the end it will be necessary to press <CTRL> <SPACE> to BREAK the QL followed by "list" or "run".

To transfer direct to QL storage - there may be simpler ways but the following works:

```
on the QL:   10 OPEN £12, ser2c
              20 OPEN £13, mdv1_filename
              30 REPEAT BBC_QL
              40 INPUT £12, inline$
              50 if inline$ = ">" then exit BBC_QL
              60 PRINT £13,inline$
              70 end REPEAT BBC_QL
              80 CLOSE £12
              90 CLOSE £13
              99 STOP
```

When the BBC has finished listing, the > prompt is displayed. Press RETURN and this will be sent to the QL, closing the serial interface at the QL. So the complete BBC type-in (including RETURNS) is:-

```
*FX 5,2 <RTN>
```



```
*FX 156,82,0 <RTN>  
LOAD "program" <RTN>  
<control - 8> LIST <RTN>  
<RTN>  
<control - C>
```

To transfer a program from QL to BBC (e.g. to store it on BBC disks):

```
on the BBC:      *FX 156,82,0  
                 *FX 2,1
```

```
on the QL:      save ser2c
```

The program will be transferred, but the BBC will be left with a dead keyboard.

Either

```
on the BBC:  <BREAK>  
             OLD  
             SAVE "newfile"
```

Or

```
on the QL:   new  
             10 a$ = "**FX 2,0"  
             20 OPEN £12, ser2c  
             30 PRINT £12, a$  
             40 CLOSE £12  
             Run
```

which should restore the BBC keyboard, enabling the program to be SAVED, now, in the usual way.

Note that the BBC <BREAK> does not restore the serial baud rate to the default value, so that if this has been altered, e.g. for a serial printer, then it will be necessary to reset it by either powering the BBC off and on. or by \*FX 7,7 and/or \*FX 8,7.

N.B. Please read HASH for £ throughout.

### **MORE ON THE EPSON RX-80 SWITCH SETTINGS.**

In the November issue of IQLUG, David Hughes gave a list of RX-80 switch settings which he got off of Rob Sherratt. I would like to add a further modification of the settings:-

```
DIP switch 2  
 1 ON  
 2 ON  
 3 ON (was previously OFF)  
 4 OFF
```

This alteration now causes the Epson to do an automatic linefeed after a carriage return command has been sent. My reason for altering this is that a problem came to light when using the previous setting. When any print or list commands are issued by the QL's BASIC it only uses a solitary carriage return which causes the output to appear on one line at the printer - not very useful! The above corrects this but the printer driver routines for the application packages have to be altered to remove the now superfluous line-feed.

Unfortunately now another problem occurs, but only with the EASEL package, which does not have a separate printer driver. When trying to use the EASEL print graph command, the manual says "Choose the printer type you are using from the list', mine (ver 1.0) Just leaps straight into the graph. The program itself generates linefeeds which are now not needed, so I have found a way to remove these permitting correct pretty graphs.

To do this you will need the additional commands "SET\_POS, BGET, BPUT" from the Library Cartridge 1. The procedure is as follows:-

1. Install the additional commands.
2. Write Enable your back-up copy of EASEL.
3. Place your back-up copy of EASEL into mdv2.
4. Enter: "open#3,mdv2\_gprint\_prt".
5. Enter: "set\_pos#3,200".
6. Enter: "bget#3,a", then "print a". This should be '10'.
7. Enter: "bput#3,00", then "close#3"

What you have done is to patch the EASEL printer driver, putting '00'(null) in place of '0A'(linefeed). You should now be able to fully use this excellent little printer for anything! If you do have a problem, drop me a line and I will try to give you a hand.

Ken Huber,  
36 Hayes End Drive,  
Hayes End,  
Middlesex. UB4 8HD.

### USING QUILL

A member rang with a couple of questions about Quill. Firstly. how can one remove all Page breaks. Secondly. how can one remove a forced Page break, since the manual suggestion of <SHIFT> -> does not work. After a bit of head scratching I came up with the following solutions.

To remove all Page breaks (except forced ones) use the DESIGN command, and set the page length to 0.

Forced Page Breaks may be removed by placing the cursor on the page break (this must be done from underneath the break line, as otherwise the cursor just skips over it) and type <CTRL>->. Whatever you do, don't type <CNTRL> ↓ as this will cause a complete crash, and you will lose all your input: If you have saved the file it will probably be alright.

The above works fine on Quill version 1.03 ADB. which I think is the current version. I don't know about other versions.

Leon Heller

### QLMON

Version 1.3 of QLMON. my monitor/debugger utility, is now available from the software library. It does not really need any documentation, as a help facility is built-in. I am not charging anything for it. but am requesting that anyone who likes it, and thinks it worth the money, sends me one pound. It may also be freely given away to anyone, whether they belong to IQLUG or not. This is an experiment on my part, to see whether this form of software distribution is viable. If it works. it might be a Good way of preventing software piracy.

I am sending a copy of the program to Popular Computing Weekly, for them to review. They seem quite keen on the idea, and it should get us some useful publicity.

Since, in conjunction with another member, I shall be selling conventionally a considerably enhanced version of QLMON, with a disassembler built-in, and lots of other goodies, I am unlikely to be enhancing the 'free' version to any great extent, but new versions will be made available from time to time, through the library.

Leon Heller

## QLKIT ANNOUNCED

Sinclair have announced that they will shortly be selling a toolkit for the QL, called QLKit for just under £25. Mostly written by the ubiquitous Tony Tebby, QLkit adds nearly 60 new functions to the QL, and includes most of the features requested by members in the survey we conducted a few months ago, including a SuperBASIC full-screen editor. I have a beta-test copy of QLkit, and it is most impressive. Some of the more interesting additions are:-

- extended EXEC commands, allowing Jobs/programs to be chained
- PIPES. for communication between programs
- prioritising of jobs
- file spooler
- job control
- direct (random) file handling
- unformatted I/O
- new file open and enquiry functions
- file maintenance (rename, view, truncate etc.)
- conversions (binary-hex etc.)
- window control
- alternative fonts
- memory management (FREE\_MEM gives the amount of free memory)

Several useful 'filter' programs and other utilities are also included, making it very good value for money.

I wouldn't be surprised if all the additional QLkit features found their way into the QL ROMs before long.

Leon Heller

## METACOMCO BCPL COMPILER

I have been using a beta-test copy of Metacomco's BCPL compiler for several weeks, and a brief review might be of interest.

BCPL is a compiled language, which is intended mainly for systems programming (writing operating systems, compilers, text editors and so on), but which is also quite suitable for writing games and applications software. Being very close to the machine, like C (which is based upon BCPL.), BCPL is a viable alternative to assembly language, for programs that need to be efficient, both in terms of memory usage and execution time. BCPL stands for Basic Combined Programming Language, by the way. The language takes a bit of getting used to, since the only form of storage is the word (32-bits in this implementation), and the only data structure is the vector (like a single dimensional array of words). This might seem a bit limiting, but it gives the programmer a tremendous amount of flexibility and freedom as to how he or she writes a program.

Some implementations compile to an intermediate form of code, which is then interpreted like some Pascals, but this compiler translates the intermediate code into 68000 native code, resulting in somewhat large, but fast, object code files. EXECable files are produced, enabling concurrently executing programs to be written.

The package comprises a cartridge, containing the compiler (in several modules, which are overlaid, to get it all in), a rudimentary linker, and a text editor (identical to the one supplied with the Metacomco assembler), together with a manual. The documentation is adequate, provided the user has a copy of Richards and Whitby-Strevens classic book on the language, and is quite easy to follow. Several useful examples are provided, and most users should be able to get the hang of the language without too much trouble.

The compiler seems quite fast and gives very informative error messages when it finds something wrong with the source code. If a run-time error occurs, a back-trace is displayed to help the programmer find out what went wrong. Considerable skill is needed to interpret the back-trace information, in some cases.

A quite comprehensive library is provided as part of the compiler, which gives access to many of the QDOS functions, including file handling (sequential and random access), screen operations, floating point, etc. Several useful functions are missing, however, and I have had to code these in assembly language. Two which spring to mind are routines enabling me to access the BEEP and KEYROW functions, and these will be placed in the IQLUG library. The user may create other libraries, which may be linked with programs that have been developed. BCPL encourages modular program development, and I have found that ones own libraries may be concatenated using a simple SuperBASIC program to create a more comprehensive library. Unfortunately, the linker provided is rather primitive, and includes the whole library, whether modules are needed or not, resulting in a large overhead at run-time of at least 9K. The linker does not always indicate if any references are unresolved, which generally results in a run-time abort of the program.

Although I still prefer C as a programming language, until a decent C compiler comes along I shall be quite happy with this implementation of BCPL. I have used it to write various utilities such as a program to convert files from upper-case to lower-case and vice-versa, which will shortly be placed in the library, in both source code and object code form. We intend to maintain a BCPL sub-section in the library, so send in your contributions to Rob.

At £59, I consider that this package is excellent value for money, and makes the Metacomco assembler look a bit overpriced, since it sells for the same amount. You can buy both packages together for £100, which looks like quite a good deal, especially as it is relatively easy to add assembly language routines to BCPL programs. If you feel that £59 is a bit expensive. you can console yourself with the thought that BCPL for the Beeb costs well over £100, and is a much inferior implementation.

Leon Heller

### **CASE CONVERSION PROGRAM WRITTEN IN BCPL**

Readers unfamiliar with BCPL might find the following program of Interest. It converts files from lower-case to upper-case and vice-versa.

```
Get "libhdr"
Get "mylibhdr"
manifest $(length = 20 $)

let start() be
$(
  let source.file.name = vec length
  let dest.file.name = vec length
  let ch = ?
  let stream.in1, stream.out1, stream.out2, error = 0, 0, 0, 0
  let stream.in2 = 0
  let option = ' '

  stream.in2 := findinput("con_")
  cls()
  screen(screen.border, 1, 7)
  writes("Convert file to upper- or lower-case*n*n")
```

```
writes("Source file: ")
readstring(source.file.name, 30)
writes("Destination file: ")
readstring(dest.file.name, 30)
writes("***n1. Upper- to lower-case")
writes("***n2. Lower- to upper-case")
Newline()
stream.in2 := selectinput(stream.in2)
$(writes("***nSelect option (1 or 2): ")
    screen(screen.cursor)
    option := rdch()
    wrch(option)
    if option = '1': option = '2' break
$) repeat

cls()
stream.in1 := findinput(source.file.name)
if stream.in1 < 0 then open.error()
stream.out1 := findoutput(dest.file.name)
if stream.out1 < 0 then open.error()

stream.out2 := findoutput("con ")
if stream.out2 < 0 then open.error()
stream.in1 := selectinput(stream.in1)
if stream.in1 = 0 then rw.error()
$( test option = '1' then ch := lowerch(rdch())
  else ch := capitalch(rdch())
if ch = endstreamch break
stream.out1 := selectoutput(stream.out1)
if stream.out1 = 0 then rw.error()
wrch(ch)
stream.out2 := selectoutput(stream.out2)
if stream.out2 = 0 then rw.error()
wrch(ch)
$) repeat
Error := endread()
Error := endwrite()
$)

and open.error() be
$)
    writes("***nFile open error")
    stop(0)
$)

and rw.error() be
$(
    writes("***nFile read/write error")
    stop (0)
$)

and lowerch(ch) = valof
$(
    if 'A' <= ch <= 'Z'    resultis ch + ('a' - 'A')
$)
```

Leon Heller

## BEGINNERS SECTION

Write a program to solve the following problem:

Find the value(s) between £1.00 and £10.00 which meet this condition:

- Exchange £ and pence figures
- Deduct £1.12
- The result must be twice the number you started with.

For example, if we start with £6.14, the 'exchange' gives us £14.06. Deducting £1.12 gives us £12.94, but this is not exactly double £6.14, though it's not far off!

There's a prize for the first correct answer (with program listing) arriving after the publication of QUANTA! I have two different programs - one algebraic, one pure logic - and I am interested to see how creative QL Users are! How about including a timing device to see how long the programs take to run?

I will provide listings for publication in a subsequent issue.

Andrew Risby  
7, Combers,  
Balcombe,  
West Sussex.  
RH17 6NE

Readers interested in the Beep command might like to try the following program which makes a passable imitation of a modern telephone. Anyone out there produced a musical note table yet?

```
10 For n 1 to 9
20 BEEP 7200, 12, 5, 10, 7, -5
30 PAUSE 25
40 BEEP 7200, 12, 5, 10, 7, -5
50 PAUSE 120
60 END FOR n
```

Douglas Hopwood,  
3 Coombe Drive,  
Addlestone,  
Weybridge,  
Surrey. KT15 1DB.

The User Guide suggests that Procedure names should be in upper case. This is good advice as it makes for more readable programs. I note that some of the library contributions do not adopt this practice. It may be due to the apparent difficulty of altering to upper case if the name was entered in error in lower case first.

It is easy to fix:-

1. Note the names to be changed.
2. SAVE the program on Microdrive.
3. Enter NEW or reset the computer.
4. Enter a new program line, say line number 1, containing each of the names in upper case (Caps lock ON) separated by colons (:).
5. MERGE the original program.  
All occurrences of the names will now be in upper case.
6. Delete line 1.
7. Re-save the program. I always include a procedure named RESAVE.

John Tanner, 43 Northumbria Drive, Bristol. BS9 4HL. 0272-623401

P.S. We have started a local Members group in Bristol.  
Any members who wish to meet should contact me.  
100 REMark insertion sort  
110 REMark written by Leon Heller

```

120 REMark translated directly from a Pascal program
130 REMark into SuperBASIC
140 :                               330     END IF
150 biggest=0:b=0                 340     END FOR i
160 length=20                     350     swap a(pointer),a(location)
170 DIM a(length)                 360 END FOR pointer
180 CLS                            370 :
190 :                               380 REMark print sorted array
200 REMark generate array of random numbers 390 PRINT
210 FOR i = 1 TO length           400 FOR i=1 TO 20
220   a(i) = RND(20)              410   PRINT a(i); " ";
230   PRINT a(i); " ";           420 END FOR i
240 END FOR i                       430 :
250 :                               440 DEFine PROCedure swap(x,y)
260 REMark sort array             450   LOCAL temp
270 FOR pointer = 1 TO length-1   460   temp=x
280   big=a(pointer):location=pointer 470   x=y
290   FOR l = pointer+1 TO length 480   y=temp
300     IF a(l)>big THEN          490 END DEFine swap
310     big=a(l)
320     location = l

```

### BIRMINGHAM AREA SUB-GROUP

The first meeting of the Birmingham Area Sub-group was held at "The Holloway" pub, Holloway Head, Birmingham on Thursday 22nd November. Despite the foul weather 17 people turned up, which was quite promising considering that the IQLUG Archive file contains only about 22 members for this region.

During a lively meeting Charles Dillon gave an excellent off-the-cuff presentation on Archive, demonstrating how its great power and possibilities could be the reward of some in-depth study of programming procedures.

There was much copying of library tapes #1 & #2 when Rob Sherratt's multi-tasking fast-copy utility proved vital to provide sufficient duplicates before closing time!

Overall it was a good start to the group which we trust will be continued at the next meeting which has been set for Monday 10th December 7.30 pm at the same venue. Those members with a B.WS,DY,WV or DY postcode will be contacted, but all others will of course be welcome.

Andrew Atton. 57 NiaeI Avenue. Birminaham. B31 1CC.

Have a Quite Lovely Christmas

<p>Microvitec Colour Monitor with cable for QL <b>£265</b> FULLY INCLUSIVE</p> 	<p>Sinclair QL Computer  <b>£389-99</b> FULLY INC.</p>
<p>QL Centronics printer interface <b>£37</b></p>	<p>Pack of 4 microdrive cassettes in wallet <b>£19</b> QL Toolkit <b>£14-95</b></p>

PRICES INCLUDE VAT & DELIVERY



(0270)  
582301

**57 REPTON DRIVE  
HAGLINGTON CREWE CW11 9AA**

Please send — QL computer(s),  
— QL cub Monitor(s) and  
— QL Centronics Interface(s).  
I enclose a cheque £ \_\_\_\_\_

NAME : \_\_\_\_\_  
ADDR. : \_\_\_\_\_

